

---

This is the **published version** of the bachelor thesis:

Moreno Barradas, Joel; Benavente Vidal, Robert, dir. Plataforma per a l'assistència en la correcció d'exàmens : reconeixement de text manuscrit. 2021. (958 Enginyeria Informàtica)

---

This version is available at <https://ddd.uab.cat/record/248440>

under the terms of the  license

# Plataforma per l'assistència en la correcció d'exàmens: Reconeixement de text manuscrit

Joel Moreno Barradas

**Resum** — En aquest projecte s'ha desenvolupat el mòdul de reconeixement de text manuscrit de la plataforma per l'assistència en la correcció d'exàmens, un projecte del Departament de Ciències de la Computació de la UAB que proporciona diverses eines de gestió i avaluació. A partir d'un arxiu PDF amb tots els exàmens escanejats d'un grup d'alumnes, el sistema de processament d'exàmens manuscrits implementat és capaç d'agrupar correctament les imatges corresponents a la resolució d'un alumne i aconseguir reconèixer correctament el número identificador de l'alumne amb una alta fiabilitat i amb un bon temps de resposta sempre que l'identificador es trobi sobre una línia horitzontal en la primera pàgina de l'examen. Per aconseguir aquests resultats s'han utilitzat les imatges de la base de dades MNIST per entrenar una xarxa neuronal LeNet-5 modificada i s'ha realitzat un fine-tuning amb imatges de dígit europeus. A més, s'ha generat un conjunt de test variat que conté diferents models d'examen omplerts manualment amb diverses cal·ligrafies i colors de bolígraf. Aquest conjunt de test ha sigut escanejat utilitzant diferents configuracions i queda a disposició de la comunitat juntament amb el codi font del sistema.

**Paraules clau**—Intel·ligència Artificial, Aprenentatge Automàtic, Aprenentatge Profund, Visió per Computador, Reconeixement Òptic de Caràcters (OCR), Xarxa Neuronal, Correcció d'Exàmens, Exàmens Manuscrits.

**Abstract** — This project has developed the handwritten text recognition module of the platform for assistance in exam correction, a project from the UAB Computer Science Department that provides different management and evaluation tools. Given a PDF file with all the scanned exams of a group of students, the handwritten exam processing system implemented is able to correctly group the images corresponding to the resolution of a student and correctly recognise the student's identifying number with a high reliability and a good response time as long as the identifier is on a horizontal line on the first page of the exam. To achieve these results, images from the MNIST database have been used to train a modified LeNet-5 neural network and a fine-tuning has been performed with images of European digits. In addition, a varied test set has been generated containing different test models filled in manually with different handwriting and pen colours. This test set has been scanned using different configurations and is made available to the community together with the source code of the system.

**Index Terms**—Artificial Intelligence, Machine Learning, Deep Learning, Computer Vision, Optical Character Recognition (OCR), Neural Network, Exam Correction, Handwritten Exams.



## 1 INTRODUCCIÓ

**D**ONADA la situació sanitària actual i tenint en compte l'adaptació de la docència a un model semi-presencial o completament en línia, el Departament de Ciències de la Computació de la UAB proposa una plataforma web per donar suport al professorat sota el nom de "Plataforma per a l'assistència en la correcció d'exàmens".

Aquesta plataforma facilita la posada en marxa i el seguiment del model docent semi-presencial o en línia proporcionant eines de gestió i avaluació dels cursos tals com:

- Creació de cursos dins d'una aplicació web, inscrivint els alumnes que el cursen i els professors responsables del mateix, registrant la informació en una base de dades.
- Publicació d'exàmens i criteris d'avaluació a un curs.

- Processament de les resolucions dels exàmens, vinculant els fulls manuscrits escanejats amb l'autor mitjançant el NIU de l'alumne.
- Retroacció dels exàmens i publicació de la puntuació en funció dels criteris d'avaluació especificats.
- Visualització d'estadístiques sobre els exàmens, els cursos i els alumnes.

Una plataforma com aquesta aporta molts beneficis pels equips docents ja que permet gestionar el procés d'avaluació, un dels processos educatius que requereixen més temps.

En primer lloc, gràcies a les tecnologies actuals, es poden implementar sistemes de correcció automàtica en plataformes d'aquest tipus. D'aquesta manera, els professors poden estalviar molt temps a l'hora de corregir exercicis on la solució és única.

Tot i que ja existeixen sistemes de correcció automàtica d'exàmens tipus test, l'ús del reconeixement òptic de caràcters permet desenvolupar sistemes més flexibles que validin respostes numèriques escrites a mà, eliminant la

---

• E-mail de contacte: joel.morenob@e-campus.uab.cat

• Menció realitzada: Computació

• Treball tutoritzat per: Robert Benavente Vidal (Ciències de la Computació)

• Curs 2020/21

dependència d'una plantilla preestablerta. En un futur, el sistema implementat en aquest projecte podria implementar un mòdul de detecció i validació de respostes.

Aquesta plataforma també permet establir una comunicació més eficaç entre estudiant i professor i permet que els professors tinguin una visió del rendiment dels alumnes gràcies a l'especificació dels criteris d'avaluació per cada exercici, la inserció de comentaris i la visualització d'estadístiques d'avaluació.

Tenint en compte les diferents possibilitats que aporta una plataforma per l'assistència en la correcció d'exàmens, es pot dir que la seva implementació es un projecte d'innovació de gran utilitat i que es podria arribar a vendre la plataforma per una futura implementació en qualsevol centre educatiu.

Per últim, és important mencionar que en aquest treball s'ha desenvolupat el tercer punt esmentat (processament de les resolucions manuscrites dels exàmens) ja que la resta d'apartats han sigut desenvolupats i implementats anteriorment [1]. La identificació de l'autor de cada resolució és un punt clau del funcionament de la plataforma ja que és el primer pas del sistema i la seva automatització permet estalviar molta feina al professorat. Per aquest motiu és molt important que el sistema detecti i reconegui l'identificador de l'alumne de manera fiable, la qual cosa ha sigut l'objectiu principal del projecte.

## 2 OBJECTIUS

### 2.1 Objectius principals

El treball a realitzar en aquest projecte es basa en desenvolupar i implementar el mòdul de processament de resolucions manuscrites. Aquest mòdul rebrà un arxiu amb les resolucions manuscrites escanejades de tots els estudiants que han fet un examen i retornarà diferents agrupacions d'imatges.

Cada grup d'imatges es correspondrà a la solució d'un mateix alumne i estarà identificada pel NIU de l'alumne en qüestió. Dins de l'objectiu principal trobem els següents subobjectius:

1. Desenvolupar una funció que rebí un document amb les resolucions manuscrites d'un examen escanejades i que separi les pàgines del document en grups d'imatges. Cada grup d'imatges es correspondrà a la resolució d'un alumne i inclourà els fulls amb els enunciats i les solucions proposades, així com fulls addicionals si ha sigut necessari.
2. Desenvolupar una funció que reconegui text manuscrit per tal d'obtenir el NIU de l'alumne en les imatges. Amb això es podrà identificar l'autor de la resolució de l'examen i es podrà desar el conjunt d'imatges de manera que quedin associades a l'alumne corresponent.

Com s'explica en l'apartat de viabilitat tècnica, l'objectiu principal del projecte es validarà mitjançant un conjunt de test amb exàmens manuscrits. Per garantir la bondat dels resultats es generaran noms, cognoms i NIUs ficticis. A més, s'utilitzaran diverses cal·ligrafies i s'ompliran els exàmens amb bolígrafs de diferents colors.

### 2.2 Objectius secundaris

A banda dels objectius principals, hi ha una sèrie d'objectius secundaris que aporten valor afegit al mòdul de processament de resolucions manuscrites i que s'aniran implementant en funció del temps disponible. Els objectius secundaris proposats són els següents:

1. Implementar millores que permetin la llibertat parcial o total del format dels exàmens per evitar la necessitat d'utilitzar un format model d'examen concret. A més, es poden aplicar millores per fer el sistema més robust de cara problemes amb el full d'examen com arrugues, taques de tinta o desviacions/rotacions a durant l'escanejat.
2. Organitzar les funcions implementades per que siguin fàcilment integrables en la plataforma de cara a guardar cada examen en el registre de l'estudiant corresponent i revisar aquells exàmens en els que no ha estat possible identificar tots els dígitos que ha de tenir un NIU.
3. Posar el codi font del sistema i el conjunt de dades de test a disposició de la comunitat científica. D'aquesta manera es poden reaprofitar els documents generats i les funcions del sistema.
4. Afegir l'opció de reconèixer el nom de l'alumne. Tot i que no permet identificar a un estudiant de manera única, pot ser d'utilitat en cas dubte o quan hi hagi errors en el reconeixement del NIU.
5. Afegir l'opció de reconèixer resultats numèrics o en format de text. D'aquesta manera es permet implementar un sistema de correcció completament automàtica en un futur.

## 3 ESTAT DE L'ART

El reconeixement òptic de caràcters (OCR) és una aplicació de la intel·ligència artificial que consisteix en la identificació de caràcters o dígitos d'una imatge. El procés del reconeixement de caràcters passa per la binarització (conversió de la imatge a escala de grisos), la segmentació (identificació dels píxels que formen un mateix caràcter) i la simplificació dels components (ús de processament d'imatge per facilitar la segmentació i la classificació).

Durant l'última dècada s'han realitzat molts estudis en aquest camp i s'ha arribat al desenvolupament de software com "Tesseract OCR" (open-source) o "Google Cloud Vision" (comercial) capaços d'obtenir bons resultats a l'hora de reconèixer text imprès. Tot i això, existeix una tasca més complexa: el reconeixement de text manuscrit.

És per això que va començar un procés de revisió bibliogràfica per veure quins mètodes de classificació permetien obtenir millors resultats. Partint de models estadístics anteriors a les xarxes neuronals convolucionals (CNNs) com els classificadors de regressió logística (Logistic Regression Classifier) o les màquines de vectors de suport (Suport Vector Classifiers) i arribant fins les xarxes neuronals més complexes, el reconeixement de text manuscrit és un dels camps més populars en el món de la intel·ligència artificial.

Centrant-nos en el reconeixement de dígitos manuscrits, hi ha moltes bases de dades que permeten entrenar i

validar la bondat de models estadístics i xarxes neuronals però la més important és la coneguda com “MNIST” (Modified National Institute of Standards and Technology) [2]. Aquesta base de dades està formada per 60.000 imatges etiquetades d'entrenament i 10.000 imatges etiquetades de test. Combina dues bases de dades que recullen dígitos manuscrits d'estudiants de secundària i treballadors del “USCB” (United States Census Bureau).

Amb la seva publicació van començar les proves sobre quin mètode proporcionava millors resultats [3]: classificadors lineals, classificadors no-lineals, “K-Nearest Neighbors”, màquines de vectors de suport, xarxes neuronals, arbres de decisió, etc.

El major salt tant en el reconeixement de dígitos com en el reconeixement d'objectes en general va arribar amb l'aparició de les xarxes neuronals convolucionals (CNNs), dissenyades per processar imatges i identificar patrons gràfics gràcies a les capes de convolució i de “pooling”. Alguns exemples de models importants són [4]:

- **LeNet-5** [5]: arquitectura enfocada en el reconeixement de dígitos manuscrits. L'entrada són imatges de 32x32 píxels en blanc i negre (1 dimensió). Aquest input passa per una capa de convolució seguida d'una capa “average pooling”. Torna a passar per una capa de convolució seguida d'una capa d'agrupació mitjana i termina passant per diverses capes “fully connected” per arribar a la capa final amb “Softmax” com a funció d'activació. Aquesta xarxa té uns 60k paràmetres.
- **AlexNet** [6]: arquitectura similar a LeNet-5 però amb 8 capes, més filtres, capes convolucionals, max pooling, drop-out, ReLU i mètode d'ajust de pesos SGD. Aquest model està enfocat al reconeixement d'objectes i va ser el guanyador de la competició “ImageNet ILSVRC-2012” (ImageNet Large Scale Visual Recognition Competition) i compta amb aproximadament 60M paràmetres.
- **VGG-16** [7]: arquitectura enfocada al reconeixement d'objectes formada per 16 capes. L'input és una imatge RGB de 224x224 píxels i passa per 5 parells de capes convolucionals i de max pooling fins arribar a tres capes “fully connected” seguides de la capa final amb “Softmax” com a funció d'activació. Aquesta xarxa té uns 138M paràmetres.
- **ResNet** [8]: arquitectura enfocada al reconeixement d'objectes amb 100 capes molt similar a la VGG però basada en el concepte d'ometre connexions, és a dir, és capaç de connectar neurones d'una capa amb neurones d'una altra capa més profunda evitant capes intermèdies. Amb això es permet reduir el problema de la saturació (o degradació) en el rendiment de les xarxes amb moltes capes de profunditat. Aquest model va ser el guanyador de la competició “ImageNet ILSVRC-2015”.

Tot i que les eines de classificació anteriors a les CNNs més senzilles com els regressors lineals, les màquines de vectors de suport o xarxes neuronals bàsiques com la “Fully Connected Neural Network” (formada per series de capes on cada neurona d'una capa està connectada amb cada neurona de la capa següent) permeten classifi-

car les imatges d'aquest dataset amb un cert accuracy, la comunitat va anar desenvolupant i publicant arquitectures més complexes basades en les CNNs.

D'aquesta forma, aconseguint millors resultats amb cada arquitectura desenvolupada, va sorgir el rànquing de classificació de dígitos de la base de dades “MNIST”. Actualment, els cinc millors models ordenats per accuracy són els següents [9]:

1. “Branching/Merging CNN + Homogeneous Filter Capsules” amb un accuracy del 99,84% (2020).
2. “EnsNet” amb un accuracy del 99,84% (2020).
3. “Efficient-CapsNet” amb un accuracy del 99,84% (2021).
4. “SOPCNN” amb un accuracy del 99,83% (2020).
5. “RMDL” amb un accuracy del 99,82% (2018).

Observant la data de publicació dels models es pot veure que, en els últims anys, la investigació en l'ús de xarxes neuronals per reconèixer dígitos manuscrits ha tingut l'objectiu de “perfeccionar” els resultats i ha centrat els esforços en obtenir el model amb més accuracy (encara que sigui per un 0,01% de diferència amb l'actual líder del rànquing). A més, es pot veure la influència de les CNNs ja que els models esmentats es basen en aquest tipus de xarxes.

## 4 VIABILITAT TÈCNICA

### 4.1 Metodologia

La metodologia emprada en aquest projecte és “Scrum”, un marc de treball enfocat al desenvolupament àgil del software. Aquesta metodologia ha estat escollida ja que permet desenvolupar de manera incremental i eficaç mitjançant “Sprints”. A més, Scrum aporta flexibilitat en el desenvolupament i en canvis de planificació ja que permet adaptar i modificar el treball a realitzar en cada Sprint.

Per adaptar aquesta metodologia al projecte, el tutor ha assumit el rol de “Product Owner” tant en les primeres reunions, on es construeix el “Product Backlog”, com en les reunions setmanals que s'utilitzen a mode de “Sprint Review”.

### 4.2 Alternatives d'implementació

Les tasques principals per implementar el sistema de processament de models d'examen manuscrits són la segmentació del document en imatges, l'agrupació d'aquestes per correspondre a la resolució d'un alumne, la detecció del NIU per assignar l'autor dels fulls manuscrits i l'emmagatzematge de les imatges agrupades en un sistema de directoris que pugui utilitzar la plataforma web.

Respecte a la primera tasca, les diferents alternatives proposades per convertir els fulls del document PDF en imatges són:

- Mòdul “pdf2image” de Python [10].
- Mòdul “ghostscript” de Python [11].

Pel que fa a la detecció de quina successió d'imatges pertany a la resolució d'un mateix alumne, les diferents alternatives proposades són:

- Especificar del nombre d'imatges que formen la re-

solució d'un alumne mitjançant un paràmetre o argument.

- Reconeixement del camp d'introducció de nom i/o NIU.
- Identificació de la capçalera de l'examen a partir d'una versió en blanc de l'examen.
- Reconeixement de patrons marcant el primer full de l'examen amb un símbol específic.

En quant al reconeixement del NIU, revisant l'estat de l'art s'han pogut analitzar diferents mètodes de classificació de text manuscrit. Les diferents alternatives proposades són:

- Logistic Regression Classifier [12].
- Support Vector Classifier (SVC) [13].
- Fully Connected Neural Network [14].
- LeNet-5 [5].
- AlexNet [6].
- VGG-13 (versió reduïda de la xarxa VGG-16 [7]).
- Residual Neural Network (ResNet) [8].
- Tesseract OCR [15].
- Google Cloud Vision API [16].

### 4.3 Selecció de la solució

Després d'analitzar les diferents alternatives, s'ha arribat a la conclusió d'utilitzar el mòdul "pdf2image" de Python per transformar els fulls del document PDF en imatges ja que aquest mòdul, a diferència del mòdul "ghostscript", està dissenyat específicament per aquest tipus d'arxius i no hi ha diferència en quant a rendiment respecte a l'altra alternativa.

Pel que fa a la detecció de quina successió d'imatges pertany a la resolució d'un mateix alumne, l'alternativa escollida és la identificació de la capçalera de l'examen a partir d'una versió en blanc de l'examen. Aquesta alternativa permet detectar l'inici d'un nou examen de manera robusta i sense necessitat d'especificar el nombre de pàgines o utilitzar símbols que marquin la primera pàgina d'un examen. Per la flexibilitat respecte al model d'examen a utilitzar i la relativa senzillesa en quant al desenvolupament ja que no requereix la identificació de camps concrets en cada imatge, aquesta serà la solució a implementar.

Per últim, respecte al reconeixement del NIU, només es descarta l'ús d'eines com "PyTesseract", que no permet obtenir resultats òptims per reconèixer text manuscrit sense fer modificacions, o "Google Cloud Vision API", que té el desavantatge de ser una eina gratuïta només si es processen menys de 1.000 imatges mensuals. Tot i que el preu d'aquesta última no és elevat, és més interessant provar quina arquitectura de xarxa neuronal permet obtenir els millors resultats en el cas de reconèixer el NIU d'un alumne.

Pel que fa al regressor logístic, a la màquina de vectors de suport i a la "Fully Connected Neural Network", es descarta el seu ús per l'accuracy que acostumen a obtenir a l'hora de classificar dígit manuscrits ja que utilitzant arquitectures que parteixen de les CNN s'obtenen millors resultats sense massa esforç.

Tant AlexNet com VGG13 i ResNet són xarxes desenvolupades per treballar amb dades de la base de dades

"ILSVRC". Tot i que proporcionen un bon accuracy en la tasca de reconèixer dígit manuscrits, es descarta el seu ús en aquest projecte ja que son xarxes enfocades a problemes de reconeixement d'imatges més complexos i l'entrenament és el més lent de totes les alternatives.

Per aquest motiu, queda oberta la possibilitat d'utilitzar diferents arquitectures de xarxes neuronals (a priori CNN com LeNet-5) entrenades amb diverses configuracions i aplicant mètodes de processament d'imatge fins obtenir un resultat fiables. Per facilitar la tasca, es pot trobar el codi de les diferents alternatives en la documentació citada.

Per garantir la bondat dels resultats caldrà generar i utilitzar un conjunt de test ampli. Aquest conjunt haurà d'incloure noms, cognoms i NIUs d'alumnes ficticis a més d'utilitzar diferents cal·ligrafies i colors dels bolígrafs utilitzats per assegurar la diversitat de les mostres.

## 5 PLANIFICACIÓ

### 5.1 Tasques

Les tasques definides per complir els objectius principals son les següents:

#### Fase 1: Inici i planificació del projecte.

- **Tasca 1.1:** Definir l'abast i l'objectiu del projecte.
- **Tasca 1.2:** Cercar informació sobre l'estat de l'art i les diferents alternatives.
- **Tasca 1.3:** Definir les dates i els objectius dels sprints.
- **Tasca 1.4:** Elaborar i lliurar l'informe inicial.

#### Fase 2: Generació del conjunt de test.

- **Tasca 2.1:** Generar el llistat de noms, cognoms i NIUs ficticis.
- **Tasca 2.2:** Generar el conjunt de test.

#### Fase 3: Realització del projecte.

- **Tasca 3.1:** Separar el document en imatges.
- **Tasca 3.2:** Agrupar les imatges que pertanyen a una mateixa resolució de l'examen.
- **Tasca 3.3:** Reconèixer el NIU a partir del primer full de la resolució manuscrita.
- **Tasca 3.4:** Nomenar les imatges i guardar-les en el sistema de directoris.
- **Tasca 3.5:** Elaborar i lliurar els informes de progrés.

#### Fase 4: Test i proves finals.

- **Tasca 4.1:** Comprovar la bondat del sistema en local.
- **Tasca 4.2:** Comprovar la bondat del sistema en l'entorn de la plataforma web.

#### Fase 5: Lliurament final.

- **Tasca 5.1:** Elaborar i lliurar la documentació final.
- **Tasca 5.2:** Pujar a producció.
- **Tasca 5.3:** Tancar de manera formal el projecte.

El diagrama WBS en forma d'arbre es pot veure en l'annex A1.

Per planificar el projecte i generar el diagrama de Gantt s'ha utilitzat "Microsoft Project". La planificació de les tasques del projecte amb el camí crític marcat en vermell es pot veure l'annex A2.

### 5.2 Definició dels sprints

Com s'ha comentat anteriorment, les reunions setmanals amb el tutor s'utilitzen a mode de "Sprint Review". Les

reunions que cauen en mig d'un sprint serveixen per veure l'estat del treball a realitzar i donen la possibilitat de realitzar modificacions o ajustaments en la planificació.

Les dates dels sprints venen marcades tant pels lliuraments establerts en la planificació del TFG (els lliuraments dels informes són l'indicatiu de sprint planificat seguint aquest criteri). En el cas dels sprints on no està previst lliurar cap informe, han sigut planificats en funció del treball a realitzar. Això permet especificar quines tasques es volen realitzar en cada sprint i dona la possibilitat d'avaluar la progressió del projecte.

D'aquesta manera, tant les dates dels sprints com la descripció del treball a realitzar i els lliuraments s'especifiquen en l'annex A3.

### 5.3 Revisió de la planificació

El seguiment de la planificació durant els primers quatre sprints ha sigut correcte en quant al treball a realitzar però incorrecte en quant al temps assignat a cada tasca.

La generació del conjunt de test s'ha vist afectada per imprevistos que no han permès aconseguir el nivell de diversitat cal·ligràfica planejat però, tot i això, s'ha aconseguit un bon nivell d'heterogeneïtat en les mostres amb un retard pràcticament nul.

La resta de tasques dels quatre primers sprints han sigut desenvolupades en un menor temps del previst. Per aquest motiu, en el quart sprint també s'entrega una primera versió del sistema de reconeixement del NIU.

En comptes de replanificar els sprints, es deixa aquesta primera versió del sistema de reconeixement del NIU com un extra ja que caldrà implementar millores o solucionar els inconvenients a l'hora de detectar els NIUs.

En quant al cinquè i sisè sprint, s'han desenvolupat quatre versions del sistema de reconeixement del NIU en comptes de dues degut a que les tasques dels anteriors sprints es van finalitzar en menor temps del previst.

## 6 SISTEMA DE PROCESSAMENT DE MODELS D'EXAMEN MANUSCRITS

En aquesta secció queden reflectits els avenços efectuats en el desenvolupament del projecte i es descriu com s'ha realitzat la feina referent a les tasques definides en la fase de planificació.

Per posar en context els següents subapartats, en la figura 1 es pot observar el diagrama de flux de l'algorisme de processament dels exàmens.

### 6.1 Generació del conjunt de test

Per realitzar aquesta tasca s'han omplert manualment diferents models d'examen amb diverses cal·ligrafies i colors de bolígraf. En aquesta tasca han participat 7 persones diferents, omplint aproximadament entre 10 i 15 exàmens per persona.

Per garantir la bondat dels resultats del mòdul a implementar cal generar un conjunt de test ampli i variat. El primer pas és generar un llistat d'alumnes ficticis amb noms, cognoms i NIUs aleatoris.

S'ha utilitzat el llistat d'alumnes ficticis esmentat ante-

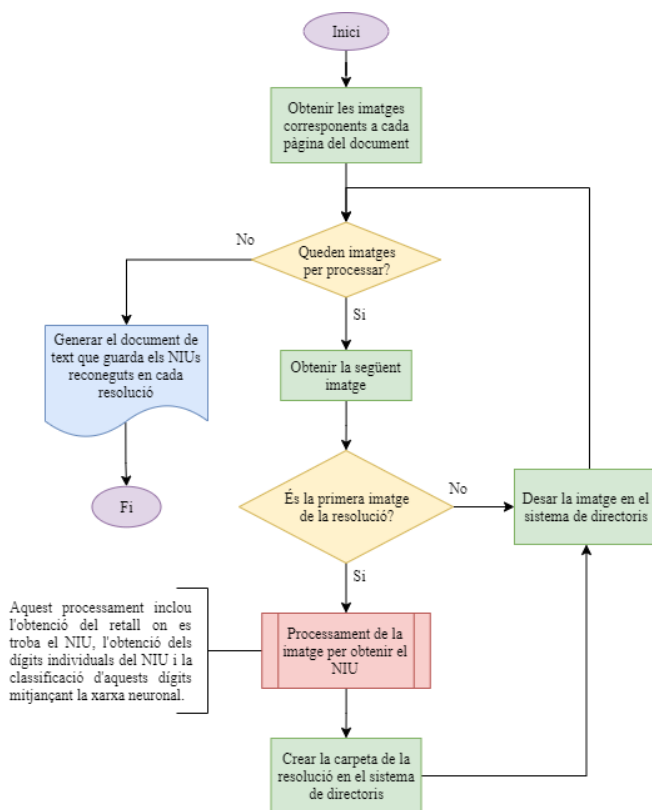


Figura 1 - Diagrama de flux del sistema

riorment per omplir els camps d'identificació de l'alumne i s'han omplert algunes preguntes amb operacions matemàtiques senzilles i dues formes d'expressar els resultats.

A més, s'han afegit exercicis realitzats en fulls apart entre les resolucions per simular el cas d'un examen real. D'aquesta manera es podrà validar la bondat del sistema d'agrupació d'imatges, que haurà de detectar quins fulls (comptant els fulls d'exercicis afegits) corresponen a la mateixa resolució.

Un cop omplerts, s'han generat quatre documents amb diferents configuracions d'escanejat per cada model d'examen. Es pot trobar una taula que descriu el conjunt de test generat en l'annex A4 i diversos exemples de pàgines del conjunt en l'annex A5.

És important destacar que aquest conjunt de test es pot aprofitar per qualsevol projecte que requereixi d'un conjunt d'exàmens resoltos a mà.

### 6.2 Separació del document en imatges

Aquesta tasca consisteix en obtenir una imatge per cada pàgina del document on es troben totes les resolucions manuals d'un examen escanejades. Això ha sigut possible gràcies a la llibreria "pdf2image" de Python i s'han utilitzat diverses configuracions fins obtenir un rendiment acceptable.

Una de les limitacions conegudes és l'ús d'aquesta llibreria amb documents extensos on finalitza el procés per problemes de memòria. Aquest problema es pot solucionar utilitzant un directori de sortida on es guarden les imatges a mesura que s'extreuen del document (en aquest cas s'ha utilitzat un directori temporal).

A més, per millorar el rendiment s'ha especificat que

les imatges es guardin en escala de grisos (grayscale = True), que tinguin format "JPEG" (fmt = "jpeg") i que s'utilitzi "Cairo", una llibreria de gràfics 2D que accelera la renderització de les pàgines del document (use\_pdftocairo = True).

### 6.3 Agrupació de les imatges

Un cop obtingudes les imatges referents a les pàgines del document cal agrupar-les en grups que es corresponguin amb una mateixa resolució de l'examen.

L'alternativa implementada es basa en detectar la primera pàgina de cada resolució utilitzant un concepte anomenat "image hashing" o "perceptual hashing", que consisteix en examinar el contingut d'una imatge i utilitzar-lo com a dades d'entrada d'una funció hash, obtenint un valor de sortida que identifiqui la imatge de manera única [17].

La llibreria "ImageHash" proporciona diferents funcions per calcular el hash d'una imatge com: average hashing (average\_hash), perceptual hashing (phash), difference hashing (dhash), wavelet hashing (whash), HSV color hashing (colorhash) i crop-resistant hashing (crop\_resistant\_hash).

Per implementar la darrera alternativa s'ha utilitzat la funció "average\_hash" d'aquesta llibreria ja que proporciona el menor temps d'execució com es pot veure en la figura 2, on s'han descartat whash, colorhash i crop\_resistant\_hash per tenir un temps mitjà superior als 300 segons.

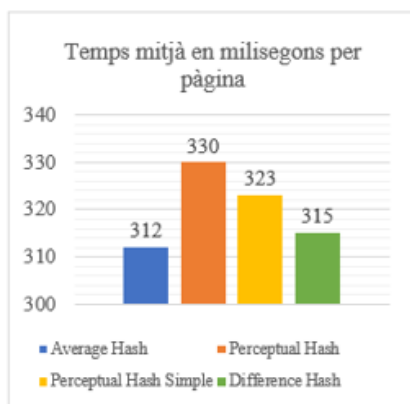


Figura 2 - Rendiment de les funcions de la llibreria "ImageHash"

Aprofitant que les imatges semblants tenen un valor de sortida similar, es pot detectar si la imatge es correspon amb el primer full d'una resolució. Només cal definir un llindar i calcular la diferència entre el valor hash de la imatge a comprovar i el valor hash de la primera imatge del document (que és el primer full d'una resolució). La imatge a comprovar es correspondrà amb el primer full d'una resolució si la diferència calculada és inferior al llindar.

Cal destacar que el sistema d'agrupació implementat agrupa les imatges correctament en un 100% dels casos tant per els models d'examen bàsics com per els models d'examen que contenen fulls addicionals.

## 6.4 Reconeixement del NIU

Durant el desenvolupament d'aquest apartat han sorgit problemes que s'han anat resolent amb diferents versions del sistema de reconeixement del NIU. En els següents subapartats es poden veure les modificacions més importants de cada versió del sistema de reconeixement del NIU.

### 6.4.1. Model 1: Model bàsic

Per reconèixer el NIU s'ha utilitzat una arquitectura de xarxa neuronal "LeNet-5" [5] modificada i entrenada amb les dades de MNIST. Les principals modificacions respecte l'arquitectura original han sigut l'addició de més capes convolucionals i l'augment del nombre de filtres, passant de 6 a 32 en les primeres dues capes convolucionals i de 16 a 64 en les següents dues capes convolucionals, la qual cosa ha permès reduir el "bias".

Com es pot veure en la figura 3, aquesta xarxa modificada consta de dos capes convolucionals seguides per una capa de "pooling" (amb 32 filtres), després segueixen dues capes convolucionals seguides d'una altra capa de "pooling" (aquest cop amb 64 filtres) i, per últim, tres capes "fully-connected" seguides d'una capa final amb funció d'activació "softmax" (10 classes, corresponents als dígit del 0 al 9).

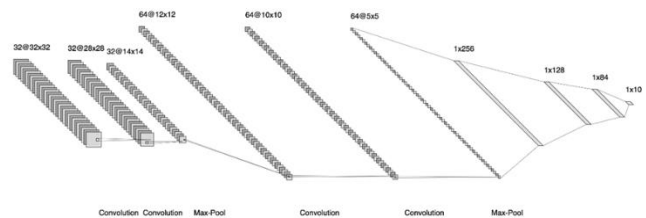


Figura 3 - Arquitectura del model "LeNet-5 v2"

Font: [18]

S'han utilitzat les dades proporcionades en la competició "Digit Recognizer" de Kaggle (que és el dataset MNIST però separat en 48.000 imatges d'entrenament i 28.000 imatges de test) i s'han estandarditzat les dades d'entrenament mitjançant la subtracció de la mitjana dividida per la desviació estàndard. Això ajuda a que l'algorisme d'aprenentatge convergeixi més ràpidament.

Per augmentar la variància, de manera aleatòria algunes imatges han sigut rotades en un rang de 0 a 10 graus, ampliades amb un factor d'ampliació de 0.1 i desplaçades de manera vertical i horitzontal entre un 0% i un 10% de l'amplada o l'alçada respectiva ment. També s'ha utilitzat una regularització amb un paràmetre lambda de 0.005 en algunes capes convolucionals, fet que ha ajudat a reduir l'overfitting, s'ha afegit un drop-out del 25% en les capes de pooling per generalitzar millor i s'ha estabilitzat la xarxa mitjançant la normalització dels outputs de les capes. Per últim, s'ha millorat el rendiment de l'entrenament mitjançant un coeficient d'aprenentatge variable.

Entrenant el model durant 30 èpoques i amb un batch size de 64, s'ha obtingut un 99,82% d'accuracy en les dades d'entrenament i un 99,62% d'accuracy en les dades de validació.



En quant a la detecció del NIU, situat en la primera pàgina de cada resolució, primer cal obtenir el retall on es troben els dígit. Això s'ha fet detectant la línia horitzontal que hi ha a sota i, un cop obtinguda la seva posició en la imatge, s'han retallat els 80 píxels que hi ha per sobre ja que els NIUs de totes les imatges es troben aproximadament en aquesta àrea (veure figura 4).

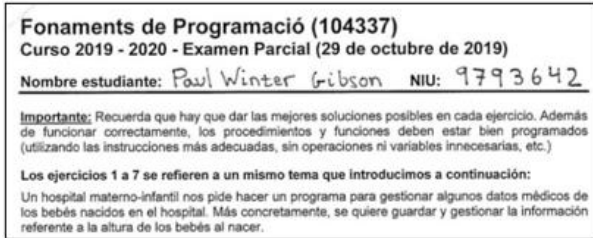


Figura 4 - Imatge d'on s'extreu el NIU

Seguidament es passa a processar el retall afegint soroll gaussià, binarització i una iteració d'erosió morfològica per reduir problemes amb els NIUs escrits amb tinta blau i millorar els resultats de les prediccions (veure figura 5).



Figura 5 - Retall del NIU processat

Per obtenir els números individuals i fer la predicció de cada un amb el model entrenat, s'ha utilitzat la implementació de l'algorisme "Canny" de OpenCV. Un cop obtinguda la imatge amb les vores ressaltades s'ha passat a utilitzar la funció "findContours" de OpenCV i les funcions "grab\_contours" i "sort\_contours" de la llibreria "imutils" per obtenir i ordenar els retalls que representen cada dígit de la imatge processada.

Aquestes imatges, que es corresponen als dígit, es tornen a processar per adaptar-se a l'input del model entrenat (imatge binaritzada de 32x32 píxels) i es tornen a estandarditzar com s'ha fet amb les dades d'entrenament.

Per últim es prediu la classe de cada imatge i es guarden aquestes etiquetes en una llista, on la concatenació de les 7 últimes posicions (els 7 últims dígit detectats) serà el NIU retornat per la funció implementada.

D'aquesta manera s'eviten errors de casos on el sistema ha identificat un caràcter situat a l'esquerra del NIU com un dígit del NIU.

#### 6.4.2. Model 2: Millora en l'obtenció del NIU

Per la segona versió del sistema de reconeixement del NIU s'ha millorat tant l'obtenció de la zona on es troba el NIU com l'obtenció de cada dígit individual del NIU.

A més, s'ha afegit soroll gaussià i s'han processat els casos on la morfologia no era suficient per tractar problemes de tinta com es pot observar en la figura 6 (a), on el número 3 queda partit per la meitat. En aquests casos es calcula la caixa delimitadora (bounding box) que conté les dues parts com es pot veure en la figura 6 (b).



Figura 6 - NIU amb el penúltim dígit partit per la meitat (a) i la detecció del mateix NIU amb el model 2 (b)

En la figura 7, que mostra la matriu de confusió entre els dígit identificats per la xarxa i els dígit reals, es pot veure que el model té problemes a l'hora de classificar el dígit 1, factor que limita l'accuracy del sistema.

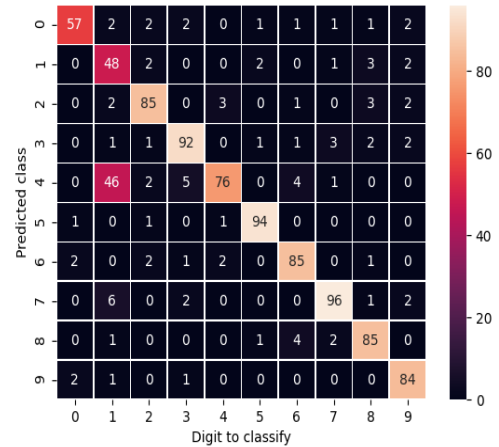


Figura 7 - Resultats de la classificació del model 2

#### 6.4.3. Model 3: Fine-tuning

La base de dades MNIST conté mostres manuscrites per estudiants d'Estats Units on alguns dígit s'escriuen de forma diferent a com es fa en altres països. Per exemple, el dígit 1 s'acostuma a escriure com es representa a la figura 8 (a) mentre que a Europa s'acostuma a escriure el dígit 1 com es representa a la figura 8 (b).

Per aquest motiu, en aquesta versió s'ha decidit realitzar un fine-tuning a les capes que realitzen la classificació (les quatre últimes capes [19]) mitjançant dígit d'una base de dades anomenada "Numbers" [20].

Aquesta conté números escrits per persones de Suïssa i es va crear amb l'objectiu de tenir una base de dades de dígit manuscrits similar a MNIST però amb números que segueixen la representació europea.

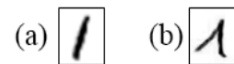


Figura 8 - Dígit 1 de la base de dades "MNIST" (a) i dígit 1 de la base de dades "Numbers" (b)

En la figura 9, que mostra la matriu de confusió entre els dígit identificats per la xarxa i els dígit reals, es pot veure que el fine-tuning realitzat permet una millor classificació del dígit 1.

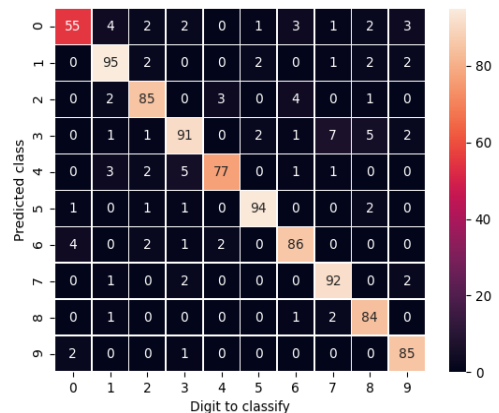


Figura 9 - Resultats de la classificació del model 3



#### 6.4.4. Model 4: Millora en l'obtenció dels dígit individuals del NIU

Revisant els casos on el model anterior no reconeix correctament el NIU s'han trobat dos problemes.

El primer, com es pot veure a la figura 10 (a), es basa en que el sistema reconeix altres elements com el número del grup o el curs i els tracta com si fossin dígit del NIU (número 43 que es correspon amb el grup).

El segon, que també es pot observar a la figura 10 (a), es basa en que dos dígit sense separació són tractats com un únic dígit (dígit 6 i 9 del NIU).

Tal i com es pot veure en la figura 10 (b), aquests problemes s'han aconseguit reduir tenint en compte la posició de les caixes delimitadores (totes les que estan per sobre del niu s'ignoren) i tenint en compte la relació d'aspecte de les caixes delimitadores (dividint la caixa delimitadora en el cas que l'amplada sigui més gran que 1.5 vegades l'alçada). Tot i això, aquestes millores no són perfectes i els errors es segueixen donant en alguns casos.



Figura 10 – Exemples d'errors en l'obtenció dels dígit individuals del NIU (a) i la detecció del mateix NIU amb el model 4 (b)

## 7 RESULTATS

### 7.1. Bondat del sistema

Com s'ha vist en l'apartat del desenvolupament del projecte, s'han implementat diverses millores fins obtenir una versió del sistema amb una bondat acceptable.

Com es pot veure en la figura 11, el salt més significatiu en el rendiment del sistema es dona entre la versió 2 i la versió 3 a causa del fine-tuning realitzat, la qual cosa demostra la importància d'utilitzar un conjunt de dades d'aprenentatge que sigui prou representatiu de la distribució de dades reals sobre la que es vol aplicar la xarxa. Un cop el model classifica amb exactitud, l'accuracy del sistema deixa d'estar limitat per l'accuracy de la xarxa neuronal.

Tot i això, amb el model 4 s'han implementat millores que han permès obtenir un accuracy del 95.06% a l'hora de classificar els dígit individuals del NIU i un accuracy del 82.35% a l'hora de reconèixer els NIUs, és a dir, dels 136 NIUs ha detectat correctament 112.

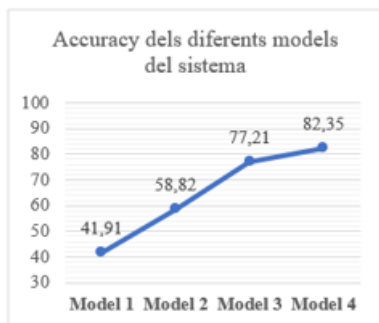


Figura 11 – Accuracy dels diferents models del sistema

Si tenim en compte l'accuracy en la classificació dels dígit individuals (95.06%) i analitzem quin hauria de ser

l'accuracy teòric a l'hora de reconèixer els NIUs, trobem aquest hauria de ser del 70.14% ( $0.9506^7$ , ja que el NIU està format per 7 dígit) i s'ha obtingut un 82.35%. Això és degut a que, en els casos on el sistema no reconeix correctament el NIU per errors de classificació, hi ha errors de classificació en més d'un dígit individual.

Analitzant els 24 casos on el model 4 no ha reconegut correctament el NIU, trobem que en 13 casos l'error prové d'una mala classificació d'un o més dígit individuals del NIU tal i com es pot observar en la figura 12 (a), en un cas s'ha produït un error a l'hora d'obtenir l'àrea del NIU tal i com es pot observar en la figura 12 (b) i, per últim, en 10 casos s'ha produït un error en l'obtenció dels dígit individuals del NIU (s'identifiquen dos dígit com un únic dígit o s'identifica un dígit com dos dígit diferents) tal i com es pot observar en la figura 12 (c) i 12 (d).

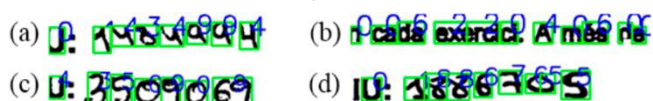


Figura 12 – Errors de classificació (a), en l'obtenció de l'àrea del NIU (b) i en l'obtenció dels dígit individuals del NIU (c) i (d)

Els casos on el problema es troba en l'obtenció dels dígit individuals del NIU no son fàcils de solucionar sense afectar negativament els casos on el sistema reconeix correctament el NIU.

Per aquest motiu, i tenint en compte que els resultats finals de la classificació són prou bons segons l'equip que està desenvolupant la plataforma de correcció d'exàmens (figura 13), es pot considerar que el sistema funciona correctament sempre i quan el model d'examen contingui la línia horitzontal sota el NIU.

El sistema es podria millorar seguint una aproximació que no tingui en compte aquesta línia horitzontal i que obtingui amb major exactitud tant el retall on es troba el NIU com els dígit individuals del NIU.

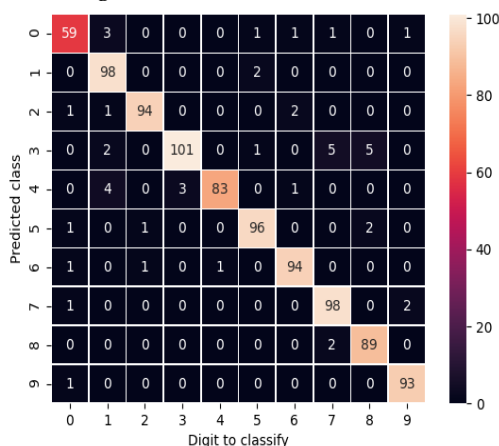


Figura 13 – Resultats de la classificació del model 4

### 7.2. Rendiment del sistema

En quant al temps d'execució, també s'han anat aplicant millores en cada versió. L'última versió processa 672 pàgines (136 resolucions diferents) en aproximadament 140 segons, és a dir, 1.03 segons per cada resolució.

Analitzant l'execució del sistema trobem, principalment, els següents processos:

- Operacions de lectura i escriptura per llegir el document, transformar aquest document en imatges i guardar les imatges.
- Processament d'aquestes imatges per obtenir el retall on es troba el NIU i els dígit individual del NIU.
- Classificació de les imatges mitjançant la xarxa neuronal.

Tal i com es pot veure en la figura 14, dels 140 segons per processar 672 pàgines el sistema utilitza aproximadament un 66% del temps d'execució en fer operacions de lectura i escriptura (92.56 segons), un 26% del temps per fer el processament de les imatges (36.23 segons) i un 8% del temps per classificar les imatges (11.07 segons).

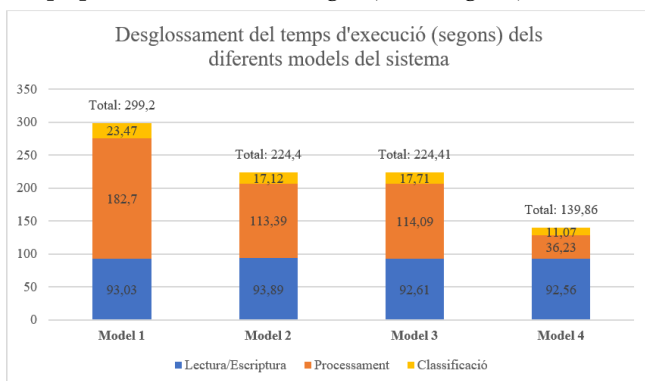


Figura 14 – Desglossament del temps d'execució dels diferents models del sistema

Per tant, tot i que en les mesures s'ha obtingut que el sistema triga 1.03 segons per cada resolució, es pot afirmar que emmagatzemant els documents i les imatges en unitats d'emmagatzematge ràpides com una unitat d'estat sòlid (SSD) es pot millorar significativament el temps d'execució del sistema.

En general, aquests resultats son força bons ja que el temps d'execució no és massa elevat i, com es pot observar en la figura 15, creix de manera lineal amb el nombre de resolucions del document.

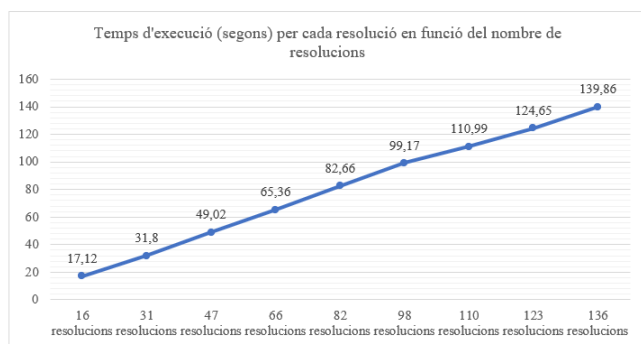


Figura 15 – Temps d'execució del model 4 en funció del nombre de resolucions

### 7.3. Codi font i conjunt de test

El codi font del sistema juntament amb el conjunt de test generat es pot trobar en el següent enllaç:

<https://github.com/joelMoreno99/tfg-exam-processing>

## 8 CONCLUSIONS

En aquest projecte s'ha implementat un mòdul de processament de resolucions manuscrites que, rebent un arxiu amb les resolucions manuscrites escanejades de tots els estudiants que han fet un examen i retornarà diferents agrupacions d'imatges (cada grup d'imatges es correspon a la solució d'un mateix alumne i s'identifica a partir del reconeixement del NIU de l'alumne).

A més, un dels punts forts del sistema desenvolupat és la seva flexibilitat, ja que l'únic requisit per garantir el bon funcionament és que el NIU es trobi sobre una línia horitzontal en la primera pàgina de la resolució.

Durant el desenvolupament del projecte s'han implementat diferents versions del sistema fins obtenir el model final, el qual aconsegueix agrupar les imatges corresponents a una resolució de manera correcta en un 100% dels casos i reconeix correctament el NIU de l'alumne en un 82.35% dels casos, un accuracy considerat com acceptable per l'equip del projecte.

Els objectius secundaris es van plantejar de manera optimista partint de la base que les xarxes neuronals convolucionals arribaven al 99% d'accuracy en la classificació de dígit de MNIST. Tot i això, s'ha aconseguit implementar un sistema relativament flexible i fàcilment integrable en la plataforma d'assistència en la correcció d'exàmens.

A més, s'ha generat un conjunt de test que, juntament amb el codi font del sistema, queda a disposició de la comunitat científica. Aquest consta de diferents models d'examen omplerts de manera manual per 7 persones diferents utilitzant diferents cal·ligrafies, bolígrafs de colors diferents i opcions d'escaneig variades.

En aquests documents s'han emplenat camps com el nom o el NIU i s'han resolt preguntes d'examen amb operacions i respostes numèriques per poder aprofitar el conjunt de test en futures ampliacions o en projectes basats en reconeixement de text manuscrit.

### 8.1. Conclusions generals del projecte

Un dels punts clau que han permès l'èxit del projecte ha sigut la definició de les tasques i la planificació dels sprints, ja que han facilitat el desenvolupament del projecte i han donat marge per implementar millores al sistema.

La revisió de l'estat de l'art i l'anàlisi de les diferents alternatives també han sigut punts destacables ja que han permès encaminar el projecte des del primer moment i no ha sigut necessari canviar les alternatives seleccionades per resoldre les tasques.

En quant a l'aprenentatge, aquest projecte ha sigut molt útil per aprofundir i posar en pràctica coneixements d'intel·ligència artificial, especialment d'aprenentatge automàtic, aprenentatge profund i visió per computador.

### 8.2. Treball futur

Tot i que es considera que el sistema funciona correctament, no és perfecte. Un dels aspectes a millorar és l'obtenció del retall de la imatge on es troba el NIU, ja que actualment es basa en trobar la línia horitzontal sota el NIU i es podria seguir un altre plantejament per fer el sistema més flexible.

També es pot millorar la detecció dels dígit individual del NIU, millorant el processament de la imatge i perfeccionant el tractament de casos anòmals com els casos on la separació entre dos dígit és insuficient.

Com s'ha esmentat anteriorment, en un primer moment es va ser massa optimista a l'hora de definir objectius secundaris. Aquells objectius secundaris que no han pogut ser desenvolupats en aquest projecte queden proposats com ampliacions o millores futures, tot i que per la seva complexitat es poden tractar com projectes independents.

## AGRAÏMENTS

M'agradaria agrair a la meua família i a la meua parella per la tota la paciència i el suport, especialment en els últims anys de carrera donada la situació sanitària tan complicada.

També m'agradaria agrair a totes les persones involucrades en el projecte per donar un cop de mà amb la generació del conjunt de test.

Finalment m'agradaria agrair al meu tutor del TFG, en Robert Benavente, per tota l'ajuda i la motivació que m'ha donat al llarg del projecte.

## BIBLIOGRAFIA

- [1] A. Rovira Solà, "Plataforma per a l'assistència en la correcció d'exàmens", Dipòsit Digital de Documents de la UAB, 2020. [Online]. Available: <https://ddd.uab.cat/record/231437>. [Accessed: 22- Feb- 2021].
- [2] Y. LeCun, C. Cortes and C. Burges, [Online]. Available: <http://yann.lecun.com/exdb/mnist/>. [Accessed: 12- Mar- 2021].
- [3] Shamim, S. M. & Miah, Md Badrul & Sarker, Angona & Jobair, Abdullah. (2018). Handwritten Digit Recognition Using Machine Learning Algorithms. Indonesian Journal of Science and Technology. 18. 10.17509/ijost.v3i1.10795.
- [4] B. Singh Bisht, "Types of Convolutional Neural Networks: LeNet, AlexNet, VGG-16 Net, ResNet and Inception Net", Medium, 2020. [Online]. Available: <https://medium.com/analytics-vidhya/types-of-convolutional-neural-networks-1enet-alexnet-vgg-16-net-resnet-and-inception-net-759e5f197580>. [Accessed: 12- Mar- 2021].
- [5] Lecun, Yann & Bottou, Leon & Bengio, Y. & Haffner, Patrick. (1998). Gradient-Based Learning Applied to Document Recognition. Proceedings of the IEEE. 86. 2278 - 2324. 10.1109/5.726791.
- [6] Krizhevsky, Alex & Sutskever, Ilya & Hinton, Geoffrey. (2012). ImageNet Classification with Deep Convolutional Neural Networks. Neural Information Processing Systems. 25. 10.1145/3065386.
- [7] Simonyan, Karen & Zisserman, Andrew. (2014). Very Deep Convolutional Networks for Large-Scale Image Recognition. arXiv 1409.1556.
- [8] He, Kaiming & Zhang, Xiangyu & Ren, Shaoqing & Sun, Jian. (2016). Deep Residual Learning for Image Recognition. 770-778. 10.1109/CVPR.2016.90.
- [9] "Papers with Code - MNIST Benchmark (Image Classification)", Paperswithcode.com, 2021. [Online]. Available: <https://paperswithcode.com/sota/image-classification-on-mnist>. [Accessed: 11- Mar- 2021].
- [10] K. Satyam, "Convert PDF to Image using Python - GeeksforGeeks", GeeksforGeeks, 2021. [Online]. Available: <https://www.geeksforgeeks.org/convert-pdf-to-image-using-python/>. [Accessed: 01- Mar- 2021].
- [11] S. Kakarla, "Using Python to Convert PDFs to Images", ActiveState, 2019. [Online]. Available: <https://www.activestate.com/blog/using-python-to-convert-pdfs-to-images/>. [Accessed: 01- Mar- 2021].
- [12] S. Mukherjee, "Recognizing Handwritten Digits with Scikit-Learn", Medium, 2020. [Online]. Available: <https://medium.com/@sharanya.mukherjee2019/recognizing-handwritten-digits-with-scikit-learn-d0700da58414>. [Accessed: 03- Mar- 2021].
- [13] D. Kothari, "SVM with MNIST", Dmkothari.github.io. [Online]. Available: [https://dmkothari.github.io/Machine-Learning-Projects/SVM\\_with\\_MNIST.html](https://dmkothari.github.io/Machine-Learning-Projects/SVM_with_MNIST.html). [Accessed: 12- Mar- 2021].
- [14] T. Bettilyon, "How to classify MNIST digits with different neural network architectures", Medium, 2018. [Online]. Available: <https://medium.com/tebs-lab/how-to-classify-mnist-digits-with-different-neural-network-architectures-39c75a0f03e3>. [Accessed: 16- Feb- 2021].
- [15] A. Chauhan, "Extracting Text from Scanned PDF using Pytesseract & Open CV", Medium, 2020. [Online]. Available: <https://towardsdatascience.com/extracting-text-from-scanned-pdf-using-pytesseract-open-cv-cd670ee38052>. [Accessed: 23- Feb- 2021].
- [16] "Detect handwriting in images | Cloud Vision API | Google Cloud", Google Cloud, 2021. [Online]. Available: <https://cloud.google.com/vision/docs/handwriting>. [Accessed: 06- Mar- 2021].
- [17] A. Rosebrock, "Image hashing with OpenCV and Python - PyImageSearch", PyImageSearch, 2017. [Online]. Available: <https://www.pyimagesearch.com/2017/11/27/image-hashing-opencv-python/>. [Accessed: 02- Apr- 2021].
- [18] J. Gupta, "Going beyond 99% - MNIST Handwritten Digits Recognition", Medium, 2020. [Online]. Available: <https://towardsdatascience.com/going-beyond-99-mnist-handwritten-digits-recognition-cfff96337392>. [Accessed: 06- Mar- 2021].
- [19] V. Roman, "CNN Transfer Learning & Fine Tuning", Medium, 2020. [Online]. Available: <https://towardsdatascience.com/cnn-transfer-learning-fine-tuning-9f3e7c5806b2>. [Accessed: 05-May-2021].
- [20] A. Schroeder, U. Beeli and M. Waligórska, "kensanata/numbers", GitHub, 2020. [Online]. Available: <https://github.com/kensanata/numbers>. [Accessed: 04- May- 2021].

APÈNDIX

A1. DIAGRAMA WBS

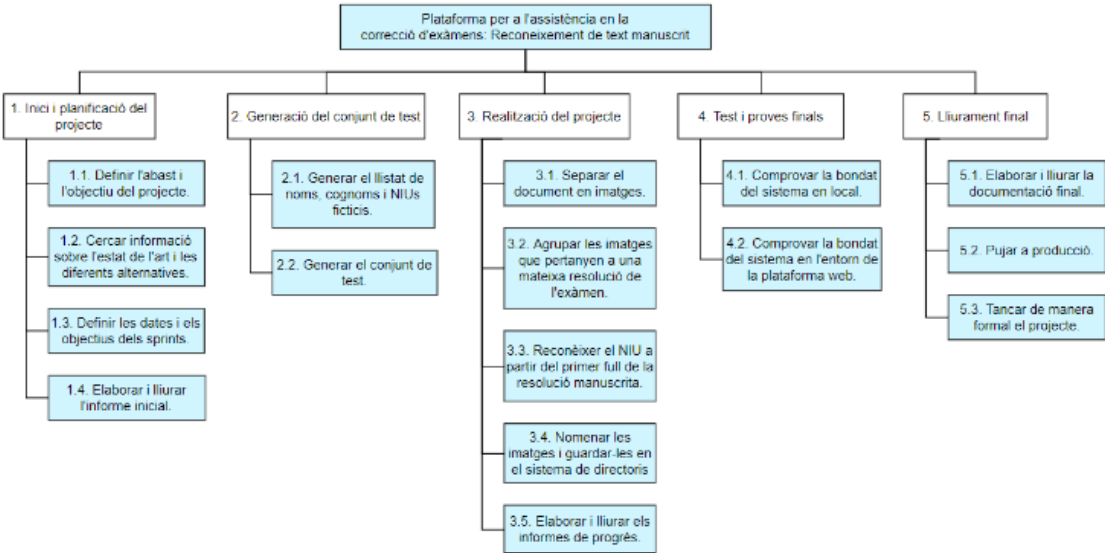


Figura 16 - Diagrama WBS

A2. DIAGRAMA DE GANTT

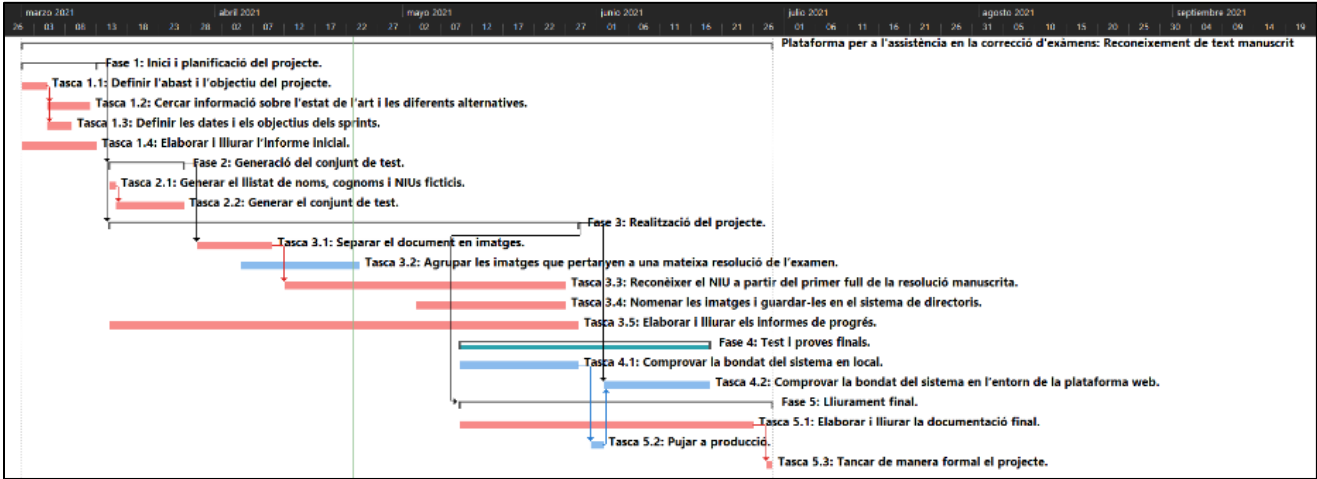


Figura 17 - Diagrama de Gantt

A3. TAULA DELS SPRINTS DEL PROJECTE

Taula 1 - Sprints del projecte

Nom	Inici	Fi	Descripció	Lliuraments
<b>Sprint 1</b>	01/03/21	14/03/21	Durant aquest sprint es pretén abordar totes les tasques de la “Fase 1”, definint l’abast i l’objectiu del projecte, l’estat de l’art, les diferents alternatives, la solució escollida i la planificació.	- Informe inicial.
<b>Sprint 2</b>	15/03/21	28/03/21	Durant aquest sprint es pretén abordar totes les tasques de la “Fase 2”, generant el llistat d’alumnes ficticis i confeccionant el conjunt de test amb resolucions manuscrites d’exàmens. A més, es començarà a redactar l’informe de progrés i es plantejarà la següent fase, fent prototips si el temps ho permet.	- Llistat de noms, cognoms i NIUs ficticis. - Conjunt de test.

<b>Sprint 3</b>	29/03/21	11/04/21	<p>Durant aquest sprint es pretén abordar la primera part de la “Fase 3”, desenvolupant el sistema que separi el document en agrupacions d’imatges que representen la resolució de l’examen d’un alumne i fent prototips del sistema d’agrupació d’imatges.</p> <p>A més, es continuarà redactant el primer informe de progrés.</p>	<ul style="list-style-type: none"> <li>- Sistema de separació en imatges del document.</li> </ul>
<b>Sprint 4</b>	12/04/21	25/04/21	<p>Durant aquest sprint es pretén finalitzar la primera part de la “Fase 3” desenvolupant i entregant el sistema d’agrupació d’imatges.</p> <p>A més, es pretén començar la segona part de la “Fase 3”, fent els primers prototips del sistema de reconeixement del NIU a partir del primer full de la resolució manuscrita.</p> <p>Aquest sprint està marcat per l’entrega del primer informe de progrés, pel que caldrà elaborar i lliurar la versió final del mateix.</p>	<ul style="list-style-type: none"> <li>- Sistema d’agrupació de les imatges.</li> <li>- Informe de progrés I.</li> </ul>
<b>Sprint 5</b>	26/04/21	09/05/21	<p>Durant aquest sprint es pretén començar la segona part de la “Fase 3”, desenvolupant la versió final del sistema de reconeixement del NIU i fent una reunió amb el “Product Owner” per definir el sistema de nomenat i desat de les imatges en el sistema de fitxers, que haurà de ser compatible amb la plataforma web actual.</p> <p>A més, es començarà a redactar el segon informe de progrés.</p>	<ul style="list-style-type: none"> <li>- Sistema de reconeixement del NIU (v1).</li> </ul>
<b>Sprint 6</b>	10/05/21	30/05/21	<p>Durant aquest sprint es pretén finalitzar la segona part de la “Fase 3”, implementant millores al sistema de reconeixement de del NIU i desenvolupant el sistema de nomenat i desat de les imatges.</p> <p>A més caldrà començar a fer les proves locals pertanyents a la “Fase 4”.</p> <p>Aquest sprint està marcat per l’entrega del segon informe de progrés, pel que caldrà elaborar i lliurar la versió final del mateix.</p>	<ul style="list-style-type: none"> <li>- Sistema de reconeixement del NIU (v2).</li> <li>- Sistema de nomenat i desat de les imatges.</li> <li>- Informe de progrés II.</li> </ul>
<b>Sprint 7</b>	31/05/21	20/06/21	<p>Durant aquest sprint es pretén finalitzar les proves de la “Fase 4” i es redactarà la primera versió de l’informe final del projecte.</p>	<ul style="list-style-type: none"> <li>- Proposta de l’informe final.</li> </ul>
<b>Sprint 8</b>	21/06/21	27/06/21	<p>Durant aquest sprint es pretén abordar totes les tasques de la “Fase 5”, deixant redactat l’informe final i recopilant els informes realitzats per entregar el “Dossier del TFG”.</p> <p>A més de la pujada a producció i el tancament formal del projecte, caldrà elaborar una proposta de presentació per la futura defensa del projecte davant el tribunal.</p>	<ul style="list-style-type: none"> <li>- Informe final.</li> <li>- Proposta de la presentació.</li> <li>- Dossier del TFG.</li> </ul>



## A4. DOCUMENTS QUE CONFORMEN EL CONJUNT DE DADES DE TEST

Taula 2 - Models d'examen escanejats

#	Model d'examen	DPI	Fulls d'exercicis	Numero de resolucions	Número de pàgines	Mida
1	FI - Nov. 2019	200	No	33	132	5,83 MB
2	FI - Nov. 2019	200	Si	33	178	7,25 MB
3	FI - Nov. 2019	300	No	33	132	9,87 MB
4	FI - Nov. 2019	300	Si	33	178	12,50 MB
5	FI - Nov. 2020	200	No	31	124	4,91 MB
7	FI - Nov. 2020	200	Si	31	196	6,85 MB
8	FI - Nov. 2020	300	No	31	124	8,39 MB
9	FI - Nov. 2020	300	Si	31	196	11,90 MB
10	FP - Oct. 2019	200	No	4	16	0,71 MB
11	FP - Oct. 2019	200	Si	4	24	0,98 MB
12	FP - Oct. 2019	300	No	4	24	1,73 MB

## A5. EXEMPLES DE PÀGINES DEL CONJUNT DE DADES DE TEST

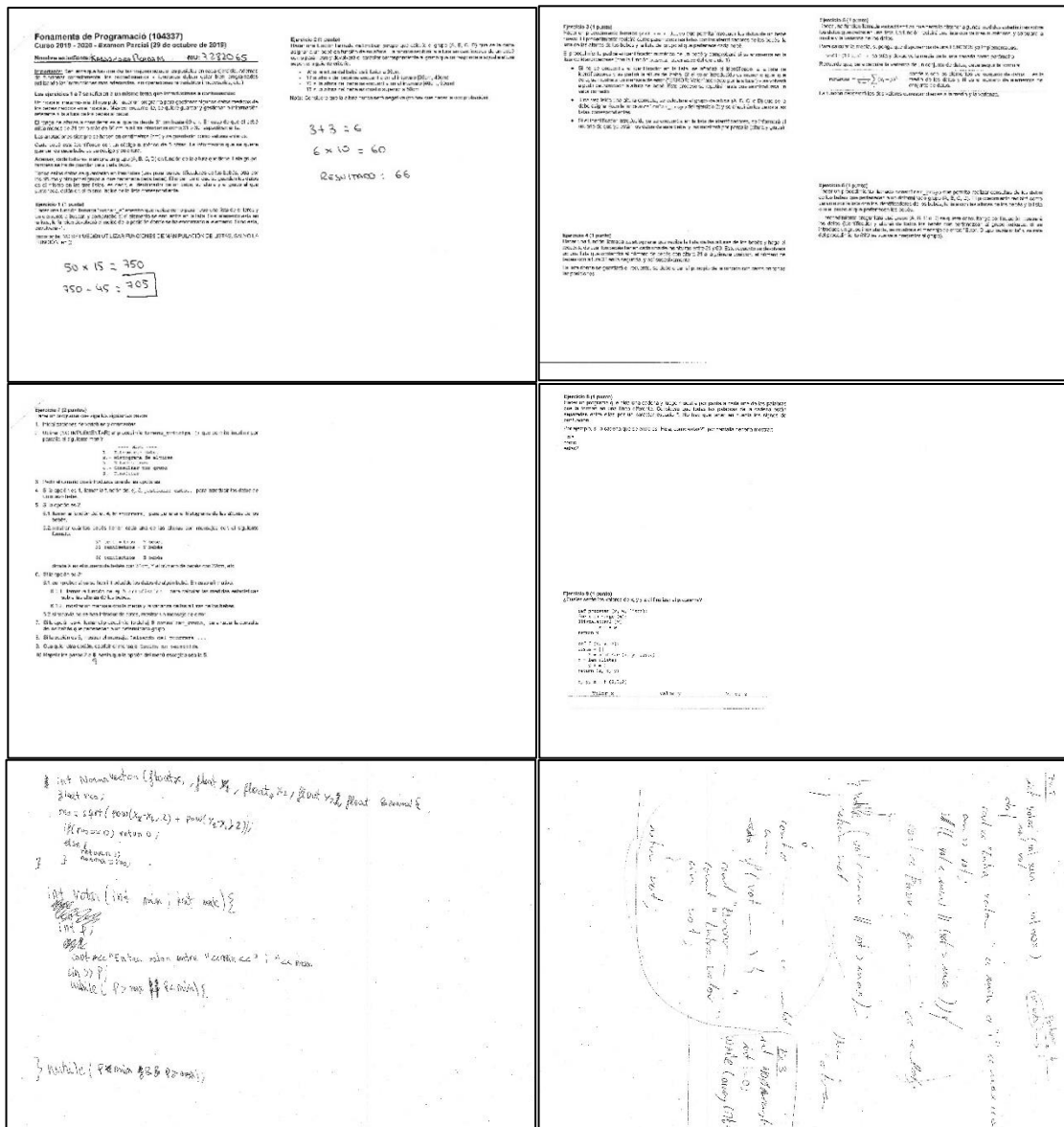


Figura 18 – Exemples de pàgines del conjunt de dades de test